

**REMARKS**

Claims 1, 2, 6, 7 and 9-31 remain pending in the application.

**Claims 1, 2, 6, 7 and 9-31 over Alam in view of Hawkins and Grambihler**

Claims 1, 2, 6, 7 and 9-31 were rejected under 35 U.S.C. §103(a) as allegedly being obvious over U.S. Patent No. 6,324,544 to Alam et al. ("Alam") in view of U.S. Patent No. 5,884,323 to Hawkins ("Hawkins"), and in further view of U.S. Patent No. 6,560,655 to Grambihler et al. ("Grambihler").

Claims 1, 2, 6, 7 and 9-31 recite, *inter alia*, a **synchronization instruction** that is comprised of at least one parameter including a control parameter identifying a **different second application to perform a next instruction after** executing the synchronization instruction.

A benefit of a **synchronization instruction** that is comprised of at least one parameter including a control parameter identifying a **different second application to perform a next instruction after** executing the synchronization instruction is flexibility. Synchronization instructions are frequently stored in a library of function calls that are called on by an application, similar to Hawkins' relationship between his sync manager library 410 and conduit library. Once an application is compiled and distributed, it is difficult to change the behavior of the application. To change which second different application performs after execution of the synchronization instruction, a programmer need only change the synchronization instruction without modifying an application that calls the synchronization instruction. This provides greater flexibility in modifying **which different second application** is called on to perform. The Examiner cited references, either alone or in combination, fail to disclose, teach or suggest the claimed features and the benefits associated with such features.

The Examiner relied on Hawkins at col. 8, lines 11-15 to allegedly teach a "SyncUnRegister() command [that] ends the synchronization and passes control to the sync manager library" that the Examiner equates to "a command that passes control to a particular application after synchronization is complete". (see Office Action, page 5)

Hawkins at col. 8, lines 11-15 teaches:

Then the conduit library ends the synchronization session by calling SyncUnRegister() function in the sync manager library 410. Control is then returned to the sync manager library 410 such that the sync manager library 410 can call another conduit library or end the synchronization process.

Hawkins discloses a conduit library that ends, with a sync manager library 410 being able to call another conduit library after an end of a synchronization session. However, Hawkins' "sync manager library 410 oversees the synchronization process and uses individual "Conduit" libraries to perform the synchronization of each database." (see Hawkins, col. 5, lines 34-37) Thus, Hawkins teaches an overseeing process, sync manager library 410, that can call another conduit library, with the conduit libraries containing the instructions for performing synchronization (see col. 5, lines 40-40-47). Hawkins' manager that calls another conduit library fails to disclose, teach or suggest a **synchronization instruction** that identifies a **different second application to perform a next instruction after executing the synchronization instruction**, as claimed.

The Office Action alleged that "Grambihler expressly discloses an instruction that includes at least one control parameter identifying a different application to perform a next instruction after executing said synchronization instruction" at col. 27, line 27 to col. 28, line 6. (see Office Action, page 5) Grambihler at the cited section teaches:

```
ISyncMgrSynchronizeInvoke::UpdateItems  
HRESULT UpdateItems( [in] DWORD dwInvokeFlags, [in]  
REFCLSID rclsid, [in] DWORD cbCookie, [in] const BYTE*  
lpCookie, );
```

Parameters

dwInvokeFlags [in] Indicates how the item should be invoked. SYNCMGRINVOKE\_STARTSYNC--When this flag is set Synchronization Manager will immediately start the synchronization without first displaying the choices to the User. SYNCMGRINVOKE\_MINIMIZED--The dialog should be initially displayed as minimized. Note: this flag will not change the state of the dialog if it already exists.

rclsid [in] CLSID of the application that should be invoked to handle the Update.

cbCookie [in] Size in bytes of lpCookie data  
lpCookie [in] Points to the private Cookie that Synchronization Manager should pass to the Initialize method of the calling application. This information can be anything the application needs such as a list of shares in the case of CSC.

Thus, a thorough review of Grambihler reveals that he at best teaches passing a Cookie to a calling application. Grambihler fails to disclose, teach or suggest a **synchronization instruction** that is comprised of at least one parameter including a control parameter identifying a **different second application** to perform a **next instruction after executing the synchronization instruction**, as recited by claims 1, 2, 6, 7 and 9-31.

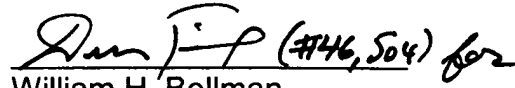
Alam, Hawkins and Grambihler, either alone or in combination, fail to disclose, teach or suggest a **synchronization instruction** that is comprised of at least one parameter including a control parameter identifying a **different second application** to perform a **next instruction after executing the synchronization instruction**, as recited by claims 1, 2, 6, 7 and 9-31.

For at least all the above reasons, claims 1, 2, 6, 7 and 9-31 are patentable over the prior art of record. It is therefore respectfully requested that the rejection be withdrawn.

**Conclusion**

All objections and rejections having been addressed, it is respectfully submitted that the subject application is in condition for allowance and a Notice to that effect is earnestly solicited.

Respectfully submitted,



William H. Bollman

Reg. No.: 36,457

Tel. (202) 261-1020

Fax. (202) 887-0336

**MANELLI DENISON & SELTER PLLC**

2000 M Street, NW 7<sup>TH</sup> Floor

Washington, DC 20036-3307

TEL. (202) 261-1020

FAX. (202) 887-0336

WHB/df